

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLPA LIMITED LIABILITY PARTNERSHIP
INCLUDING LAW CORPORATIONS

TELEPHONE (408) 720-8300

FACSIMILE (408) 720-8383

BSTZ_MAIL@BSTZ.COM
WWW.BSTZ.COM

INTELLECTUAL PROPERTY LAW

1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040OTHER OFFICES
LOS ANGELES, CA
ORANGE COUNTY / COSTA MESA, CA
PORTLAND / LAKE OSWEGO, OR
SEATTLE / KIRKLAND, WA
DENVER / ENGLEWOOD, CO**FACSIMILE TRANSMITTAL SHEET**

Deliver To: Li Zhen

Firm Name: U.S. PATENT TRADEMARK OFFICE

Fax Number: 571-273-3768 **Telephone No.:** 571-272-3768

From: Paul Krueger **Reg. No.:** _____

Date: June 12, 2009 **Time:** _____

BSTZ Matter No. 5693P286 **Number of pages including cover sheet:** 10

In Re Patent Application of: Vijayan Rajan et al.

Application No.: 09/828,271

Filed: April 5, 2001

Title: SYMMETRIC MULTIPROCESSOR SYNCHRONIZATION USING MIGRATING SCHEDULING DOMAINS

Please find attached a request for interview along with a proposed amendment for discussion purposes.

Thank you.

CONFIDENTIALITY NOTE

The documents accompanying this facsimile transmission contain information from the law firm of Blakely Sokoloff Taylor & Zafman LLP that is confidential or privileged. The information is intended to be for the use of the individual or entity named on this transmission sheet. If you are not the intended recipient, be aware that any disclosure, copying, distribution, or use of the contents of this faxed information is prohibited. If you have received this facsimile in error, please notify us by telephone immediately so that we can arrange for the retrieval of the original documents at no cost to you. IF YOU EXPERIENCE ANY DIFFICULTY IN RECEIVING THE ABOVE PAGES, PLEASE CALL (408) 720-8300 AND ASK FOR THE OPERATOR NAMED ABOVE.

Applicant Initiated Interview Request Form

In Re Application of Vijayan, Rajan et al. Attorney Docket No. 5693P286
 Application Number 09/828,271 Status of Application: Pending
 Filed April 5, 2001
 For Symmetric Multiprocessor Synchronization Using Migrating Scheduling Domains
 Group Art Unit 2194 Confirmation No. 6350 Examiner Zhen, Li B
 Proposed Date of Interview: June 17, 2009 Proposed Time: 2:00 (AM/PM) EDT
 Type of Interview Requested: (1) ☒ Telephonic (2) ☐ Personal (3) ☐ Video Conference
 Tentative Participants: (1) Joe Sosinski (2) Paul Krueger (3) Examiner Zhen (4) _____
 Exhibit To Be Shown or Demonstrated: ☐ Yes ☒ No
 If yes, provide brief description: _____

ISSUES TO BE DISCUSSED

Issues (Rej., Obj., etc.)	Claims/Figs Fig. #s	Cited Art	Discussed	Agreed	Not Agreed
(1) <u>Rej. 103(a)</u>	<u>11</u>	<u>Feltelson Article</u>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Brief Description of Arguments to be Presented:

Clarify Examiner's interpretation of prior art, including what feature of the cited reference Examiner interprets as "changing association of a task of the plurality of tasks from a first scheduling domain to a second scheduling domain, if the task requires one of the shared resources assigned to the second scheduling domain." Please see attached proposed claim amendment for discussion purposes.

shown above is reduced by half, and the resulting fee is: \$ _____

☐ A check in the amount of the fee is enclosed.

☐ The Commissioner is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. 02-2666. I have enclosed a duplicate copy of this sheet.

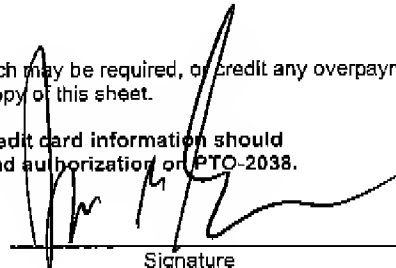
Warning: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

I am the

☐ applicant/inventor.

☒ attorney or agent of record.

☐ attorney or agent acting under 37 CFR 1.34(a).
 Registration number if acting under 37 CFR 1.34(a) _____


 Signature

Joseph W. Sosinski

Typed or printed name

62,807
 (Reg. No.)

Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest of their representative(s) are required. Submit multiple forms if more than one signature is required, see below.

☒ *Total of 1 forms are submitted

Attorney Docket No. 5693P286Patent

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application for:

Vijayan Rajan et al.

Serial No.: 09/828,271

Filing Date: April 5, 2001

For: SYMMETRIC MULTIPROCESSOR
SYNCHRONIZATION USING
MIGRATING SCHEDULING
DOMAINS

Examiner: Zhen, Li B.

Group Art Unit: 2194

Confirmation No.: 6350

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450PROPOSED AMENDMENT FOR DISCUSSION PURPOSESNOT TO BE ENTERED IN RECORD

IN THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the present application:

1-10. (Canceled)

11. (Currently amended) A method comprising:

running a plurality of tasks in a multiprocessor system that includes a plurality of processors, each processor having an identity;

scheduling the plurality of tasks using a plurality of scheduling domains by scheduling tasks on a processor independent of the identity of the processor, wherein none of the plurality of scheduling domains is bound to any one processor of the plurality of processors;

implicitly synchronizing the tasks with regard to one or more resources shared by the tasks in said system by associating said tasks with the scheduling domains, wherein each of the one or more resources is assigned to one of the scheduling domains and is not subject to domain migration, wherein implicitly synchronizing the tasks comprises

prohibiting tasks that are each associated with a same scheduling domain from running concurrently, and

allowing tasks that are each associated with different scheduling domains to run concurrently; and

changing association of a task of the plurality of tasks from a first scheduling domain to a second scheduling domain, if the task requires ~~a resource~~ one of the shared resources assigned to the second scheduling domain.

12. (Currently amended) A system comprising:

a plurality of processors, each processor having an identity;

a memory coupled to each of the plurality of processors, the memory storing data defining a set of tasks, each task of the set of tasks being runnable on more than one of said processors, each said task being associated with one of a plurality of scheduling domains, each of the plurality of scheduling domains controlling one or more shared resources, wherein each of the shared resources is not subject to domain migration; and

a scheduler to schedule the set of tasks using a plurality of scheduling domains by scheduling tasks on a processor independent of the identity of the processor, wherein none of the plurality of scheduling domains is bound to any one processor of the plurality of processors, where the scheduler prohibits tasks that are each associated with a same scheduling domain from running concurrently but allows tasks that are each associated with a different one of the plurality of scheduling domains to run concurrently, and wherein the scheduler changes association of a task of the set of tasks from a first scheduling domain to a second scheduling domain if the task requires ~~a shared resource~~ one of the shared resources controlled by the second scheduling domain.

13. (Canceled)

14. (Previously presented) A system as in claim 12, wherein at least one of the set of tasks is associated with more than one scheduling domain of the plurality of scheduling domains.

15. (Canceled)

16. (Original) A system as in claim 12, wherein said scheduler includes a plurality of runnable queues, one per scheduling domain.

17-22. (Canceled)

23. (Currently amended) A process comprising:

scheduling a plurality of tasks in a multiprocessor system that includes a plurality of processors, each processor having an identity, by scheduling tasks on a processor independent of the identity of the processor, wherein none of the plurality of scheduling domains is bound to any one processor of the plurality of processors;

performing implicit synchronization of the plurality of tasks, said implicit synchronization dividing said tasks into the scheduling domains, at least one of the scheduling domains being associated with at least two tasks of the plurality of tasks and a resource shared by the at least two tasks, wherein the shared resource is not subject to domain migration, and wherein tasks within a same scheduling domain are prohibited from running concurrently even if run on different processors and tasks that are each from a different scheduling domain are allowed to run concurrently; and

moving a task of the plurality of tasks from a first scheduling domain to a second scheduling domain, if the task requires ~~a resource~~ one of the shared resources controlled by the second scheduling domain.

24-29. (Canceled).

30. (Currently amended) A method of scheduling a plurality of processes in a multiprocessor system, the method comprising:

associating the plurality of processes with a plurality of scheduling domains wherein none of the plurality of scheduling domains is bound to any one processor in the system, and wherein each of the processes is executed by a processor independent of an identity of the processor;

implicitly synchronizing the plurality of processes with regard to one or more shared resources by prohibiting concurrently executing processes that are each associated with a same scheduling domain but allowing concurrently executing processes that are each associated with a different one of the plurality of scheduling domains, wherein each of the shared resources is assigned to one of the scheduling domains and is not subject to domain migration; and

changing association of a first process of the plurality of processes from a first scheduling domain to a second scheduling domain, if the first process requires ~~a resource~~ one of the shared resources associated with the second scheduling domain.

31. (Previously presented) The method of claim 30 further comprising allowing concurrently executing processes that are not associated with any one of the plurality of scheduling domains.

32. (Previously presented) The method of claim 30, wherein at least one of the plurality of processes is associated with more than one of the plurality of scheduling domains.

33. (Previously presented) The method of claim 30, wherein each of the plurality of scheduling domains is associated with a different one of a plurality of runnable queues.

34. (Currently amended) A method implemented in a multiprocessor system, the method comprising:

executing a software program that defines a plurality of tasks and assigns each of the plurality of tasks to one of a plurality of scheduling domains, wherein none of the plurality of scheduling domains is bound to any one processor in the system;

running a plurality of processes, each of the plurality of processes performing a different one of the plurality of tasks, wherein each of the plurality of processes is run by a processor independent of an identity of the processor;

prohibiting concurrently executing processes performing tasks that are each assigned to a same scheduling domain;

allowing concurrently executing processes performing tasks that are each assigned to a different one of the plurality of scheduling domains; and

allowing changing assignment of at least one task from a first scheduling domain to a second scheduling domain during executing the software program, if said at least one task requires ~~a resource~~ one of the shared resources assigned to the second scheduling domain, wherein the shared resource is not subject to domain migration.

35. (Currently amended) A processing system comprising:

a plurality of processors, each processor having an identity;

a memory coupled to each of the plurality of processors, the memory storing instructions which, when executed by one or more of the plurality of processors, cause the one or more of the plurality of processors to perform a method comprising:

executing a software program associating a plurality of tasks with a plurality of scheduling domains and assigning a plurality of resources to the plurality of scheduling domains, wherein none of the plurality of scheduling domains is bound to any one processor of the plurality of processors, and wherein each task of the plurality of tasks is scheduled on one of the processors independent of the identity of the processor;

prohibiting concurrently executing processes to perform tasks that are each associated with a same scheduling domain but allowing concurrently executing processes to perform tasks that are each associated with a different one of the plurality of scheduling domains; and

changing association of a first task of the plurality of tasks from a first scheduling domain to a second scheduling domain, if a process performing the first task requires a resource one of the shared resources assigned to the second scheduling domain, wherein the shared resource is not subject to domain migration.

36. (Currently amended) A computer-readable storage medium storing instructions therein which, when executed by one or more processors of a processing system, cause the one or more processors to perform a method comprising:

executing a software program that defines a plurality of tasks and assigns each of the plurality of tasks to one of a plurality of scheduling domains, wherein none of the plurality of scheduling domains is bound to any one processor in the system;

running a plurality of processes, each of the plurality of processes performing a different one of the plurality of tasks, wherein each of the processes is run on a processor independent of the identity of the processor;

prohibiting concurrently executing processes performing tasks that are each assigned to a same scheduling domain;

allowing concurrently executing processes performing tasks that are each assigned to a different one of the plurality of scheduling domains; and

allowing changing assignment of at least one task from a first scheduling domain to a second scheduling domain during executing the software program, if said at least one task requires ~~a resource~~ one of the shared resources assigned to the second scheduling domain, wherein the shared resource is not subject to domain migration.

37. (Currently amended) A method, comprising:

associating a task of a plurality of tasks with a scheduling domain of a plurality of scheduling domains, wherein the plurality of tasks share one or more resources and each of the one or more resources is assigned to one of the plurality of scheduling domains;

scheduling the task, using the scheduling domain, on a processor in a multiprocessor system that includes a plurality of processors, independent of an identity of the processor;

prohibiting tasks that are each associated with a same scheduling domain from running concurrently;

allowing tasks that are each associated with different scheduling domains to run concurrently; and

changing association of the task from a first scheduling domain to a second scheduling domain, if the task requires ~~a resource~~ one of the shared resources assigned to the second scheduling domain, wherein the shared resource is not subject to domain migration.